

УДК 681.3

ОБРАБОТКА ЗАПРОСОВ В БАЗЕ ДАННЫХ МОБИЛЬНЫХ ОБЪЕКТОВ

Д.Д. Альшаер, В.В. Губарев*

Новосибирский государственный технический университет

E-mail: jawdat_alshaer@hotmail.com

Предложен новый способ индексации отрезков траекторий мобильных объектов назван TP-деревом. Такой способ позволяет сохранить память и уменьшить время обработки запроса путем сохранения траекторий отрезков в базе данных единожды с использованием алгоритмов R⁺-дерева в отличие от R-дерева, где отрезки сохраняются и индексируются много раз (каждый раз при движении мобильного объекта на том же отрезке). Приведены результаты экспериментальных исследований, подтверждающие, что работа TP-дерева лучше, чем традиционных алгоритмов R-дерева.

Ключевые слова:*База данных, мобильные объекты, пространственно-временной запрос.*

1. Введение

1.1. Вводные замечания

Прогресс беспроводных средств связи, сенсорных технологий и GPS услуг привел к быстрой компьютеризации общества. Появилось множество новых приложений с использованием услуг определения местоположения мобильных объектов (МО) (туристические услуги в режиме реального времени, услуги транспортного и воздушного сообщения, предоставление прогноза погоды и неотложной медицинской помощи и т. д.). Все эти приложения используют базы данных мобильных объектов (БДМО).

В последнее время предпринимались многочисленные попытки индексирования позиций МО и автоматизации обработки запросов о них (см., например, [1–7]). При этом каждый мобильный объект представляется точкой, движущейся на плоскости во времени. Иными словами, описываются и запрашиваются только сведения о позиции центра объекта и не учитываются другие размерные характеристики объекта.

В стандартных базах данных (БД) данные, как правило, статичные, в то время как в БДМО хранятся сведения об отсчетах динамически развивающихся во времени не всегда хорошо предсказуемых траекторий.

Первая особенность известных методов индексации и обработки запросов о траекториях в БДМО в том, что индексация в них касается только текущих позиций МО и обработки запросов об их настоящих и будущих позициях [2], либо о прошлых позициях [3]. Однако зачастую требуются одновременные сведения о прошлых, настоящих и будущих позициях МО.

Вторая особенность известных методов индексации и обработки запросов о траекториях состоит в том, что эти методы ориентированы на работу с пространственными данными, изменяющимися непрерывно, но представленными в дискретные моменты длительного периода времени, когда не-

прерывный характер траектории не учитывается (например, модели траекторий в виде R-деревьев для мультимедийных данных [4], наложение четвертных деревьев (Quad-Trees [5]) и разновидностей R-дерева для пространственных данных [1]). При работе индексной структуры R-дерева возможна ситуация, когда в пересечении двух или более охватывающих прямоугольников содержится одна и та же запись. Когда алгоритм дойдет до выбора одной из записей, то на уровне, где прямоугольники пересекаются, произойдет ветвление вариантов поиска. Оно сильно замедлит работу поиска алгоритма в БДМО. R⁺-дерево [6] является другой версией R-дерева, которая не имеет пересечений между узлами. В [7] модель траектории в виде R-дерева была использована для индексации траекторий в многомерных методах доступа. В [7] были предприняты попытки уменьшить «мертвое» (т. е. потерянное, не допускающее восстановления) пространство путем «искусственного обновления объектов» аппроксимацией траекторий минимально ограниченным прямоугольником (МОП). В [8] описаны методы индексации на базе представления траекторий с помощью моделей ТВ-дерева и STR-дерева, основанных на делении траекторий на отрезки. Однако проблема «мертвого пространства» в них не была решена.

1.2. Постановка задачи

В цифровых БДМО мобильные объекты посылают координаты своих местоположений серверу в определенные дискретные промежутки времени. Поэтому между этими временными промежутками сервер базы данных не может выдать точное местоположение объектов.

Для предоставления сведений о движущихся объектах необходимо все возможные траектории индексировать. В настоящей работе описывается алгоритм индексирования, основанный на замене решетчатой траектории её кусочно-линейной аппроксимацией, в частности, интерполяцией, с последующей индексацией всей траектории по индексам отрезков с использованием TP-дерева.

В связи с этим целью настоящей работы является разработка такого метода индексации траекторий МО, который позволит получать из БДМО сведения о прошлых, настоящих и будущих положениях МО (будущие положения могут быть получены при использовании последней известной скорости и направления МО) с учетом непрерывности реальной траектории движения объектов. Таким методом является описываемый метод ТР-дерева.

В соответствии с поставленной целью сформулированы следующие задачи:

- аналитический обзор и сравнительный анализ существующих методов индексации МО;
- разработка нового метода, свободного от выявленных недостатков существующих методов;
- машинное имитационное исследование предложенного метода.

2. Индексация отрезков движения мобильных объектов с использованием ТР-дерева

Предположим, что пространственная БД состоит из набора кортежей, представляющих отрезки траекторий МО, двигающихся по этим отрезкам в разные временные интервалы, а каждый кортеж имеет свой уникальный идентификатор, который можно использовать для индексации и поиска МО.

ТР-дерево состоит из R^+ -деревьев. Основная идея его заключается в разделении объемной (по координатам x, y, t) траектории МО на линейные временные отрезки, пронумерованные с помощью одномерного R^+ -дерева (условно назовем его временным) и пространственного (x, y) двухмерного R^+ -дерева для индексации координат начала и конца этих отрезков. К каждому отрезку применяется одномерное временное R^+ -дерево для индексации всех объектов, двигающихся по этим отрезкам в разные временные интервалы (см. рис. 1).

Каждый листовой узел в пространственном дереве содержит индексную запись вида: (МОП,

идентификатор кортежа дорожного отрезка в БД, ссылка), где МОП записан как координаты $(x', y')(x'', y')(x''', y'')(x''', y'')$ вершин самого маленького прямоугольника, охватывающего рассматриваемый отрезок, а ссылка указывает на корень временного дерева, индексирующего моменты времени отрезка, на котором находится интересующий нас МО.

Не листовые узлы пространственного дерева содержат записи вида: ссылка к дочернему узлу (потомку), МОП, где ссылка указывает на узел низшего порядка в пространственном дереве, а МОП – координаты вершин охватывающего прямоугольника, который содержит все прямоугольники в дочерних узлах.

Каждый листовой узел временного дерева содержит индексную запись вида: ссылка на отрезок, идентификатор МО в БД, временной интервал движения (t_1, t_2) , направление движения.

Не листовые узлы временного дерева содержат записи вида: ссылка к дочернему узлу (потомку), временной интервал, где ссылка указывает на узел низшего порядка во временном дереве и временной интервал вершин охватывающего интервала, который содержит все интервалы в дочерних узлах.

2.1 Алгоритм добавления в ТР-дерево

Как следует из изложенного выше, в ТР-дерево траектории МО разделены на линии, вставлены в БД и индексированы ТР-деревом с использованием алгоритмов вставки R^+ -дерева.

Как только МО заканчивает движение на отрезке, он отправляет серверу координаты начала (x_1, y_1) и конца (x_2, y_2) отрезка, пройденного объектом, и границы временного интервала (t_1, t_2) , в течение которого он двигался по данному отрезку. Тем самым сведения об объекте добавляются в БД и индексируются ТР-деревом с использованием алгоритма добавления R^+ -дерева. Это происходит следующим образом.

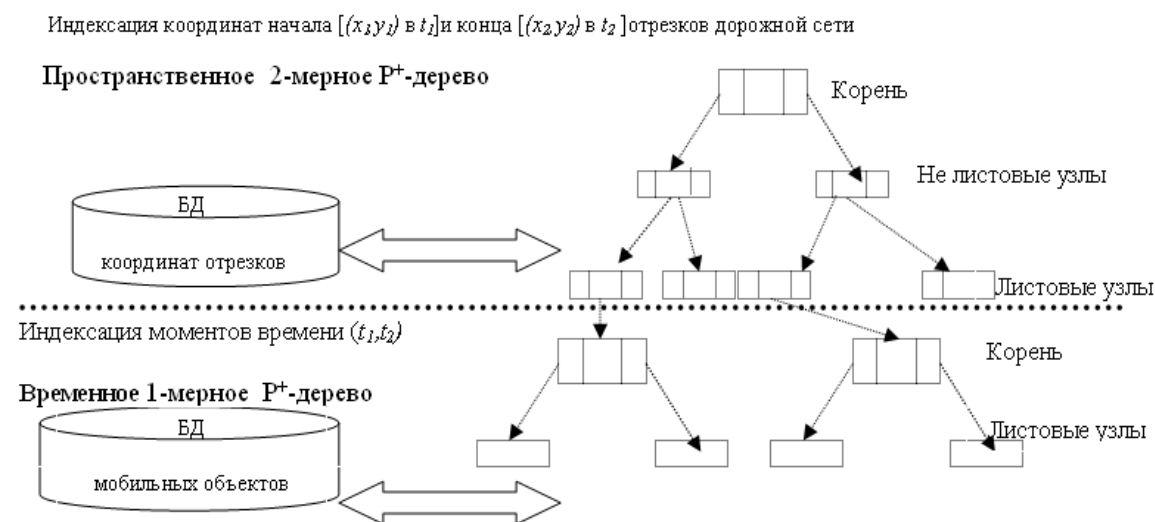


Рис. 1. Структура и принцип работы ТР-дерева

1. Для добавления записи о новом объекте вначале осуществляется поиск в пространственном дереве отрезка с координатами начала (x_1, y_1) и конца (x_2, y_2) . Как показано на рис. 2 (см. жирные линии), поиск начинается с корня дерева и идет до листового узла. В корне дерева в каждом из таких не листовых узлов выбираются те не листовые узлы, которые содержат координаты $(x_1, y_1); (x_2, y_2)$ (A на рис. 2), затем посещается листовый узел (B на рис. 2), содержащий отрезок (k), который прошел объект. Поиск идет по ссылке узла к временному дереву (C на рис. 2) соответствующего отрезка.
2. Запись о мобильном объекте O добавляется в БДМО.
3. Новая запись (D на рис. 2) добавляется во временное дерево вида: ссылка на отрезок (k), идентификатор объекта (O), интервал движения (t_1, t_2) , направление движения (0 – слева направо).

2.2 Поиск в ТР-дереве

Обработка запроса может быть реализована с помощью операции поиска. Например, «Найти

все объекты, находящиеся в определенной окрестности» или «Найти все объекты, содержащие в себе определенную область». Алгоритм поиска в ТР-дереве проиллюстрирован рис. 3. Предположим, что мы ищем объект в ТР-дереве с пространственно-временным окном запроса $(x_1, y_1, t_1; x_2, y_2, t_2)$.

1. Первый шаг поиска требует осуществления пространственного поиска (с параметрами 2-мерного интервала $(x_1, y_1; x_2, y_2)$) среди пространственного дерева, чтобы определить местонахождение отрезков линии, которые пересекают пространственное окно запроса (например, на рис. 3 это отрезки a, b, на которых находятся объекты O_1 и O_2 и отрезок e) и соответствующих временных деревьев (C и G на рис. 3) ТР-деревя (правая часть рис. 3).
2. Второй шаг требует выполнения поиска (с параметрами одномерного интервала (t_1, t_2)) в каждом временном дереве, полученном на 1 шаге. В данном примере поиск привел к узлам (G и H) временного дерева, содержащий идентификаторы МО в БД (O_1 и O_2).

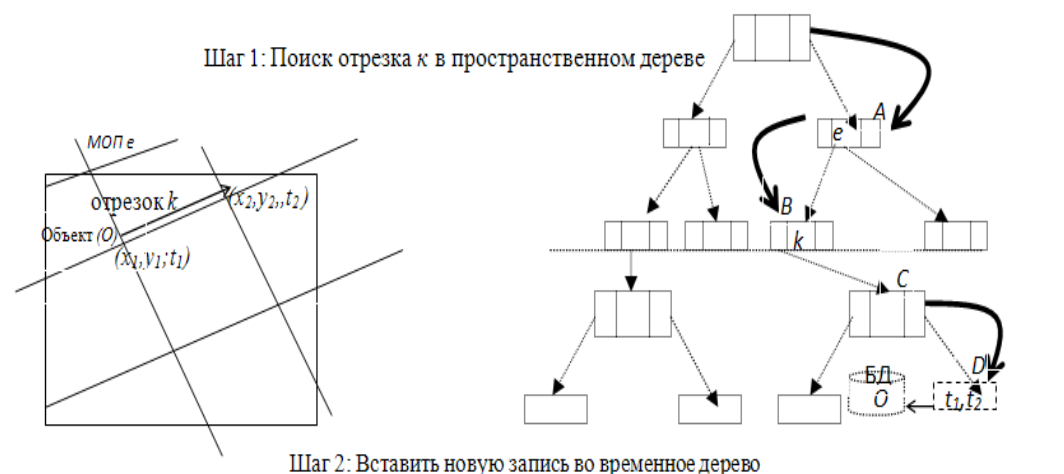


Рис. 2. Вставка объекта в ТР-дереве

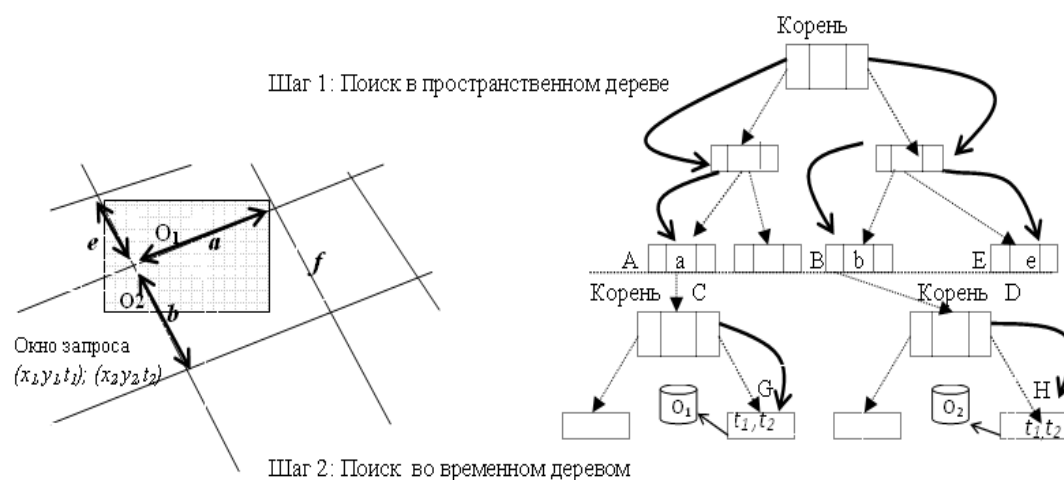


Рис. 3. Поиск в ТР-дереве

3. Имитационное исследование алгоритмов

Для экспериментальной проверки алгоритмов ТР-дерева на ПК было создано программное обеспечение, реализованное на Visual basic. Цель исследования – оценить полезность результатов, практичность алгоритмов ТР-дерева, измерить времена выполнения алгоритмов и сопоставить их с аналогичными показателями трехмерного Р-дерева.

Используемые ранее запросы касались сведений о точном нахождении объекта в заданном месте и в заданное время. В связи с этим для того, чтобы проверить полезность полученных результатов предложенной модели ТР-дерева и реализующих её алгоритмов, необходимо симитировать реальные наборы траекторий. Для этого, используя генератор сети из [9], были построены всемирные дорожные сети городов Oldenbourg и San Jose*, в которых имитировалось движение числа объектов $N=100, 200, 300, \dots, 2000$ МО. Тем самым имитировались N разных траекторий движения МО. Все траектории сети были проиндексированы с использованием ТР и Р-деревьев. Целью эксперимента являлось решение двух задач. Первая задача: сравнение размера ТР-дерева, которое индексирует различные числа МО для г. Oldenbourg с размером трехмерного Р-дерева для этого же города. Вторая задача: определить как возрастет количество узлов, которые необходимо проверить при выполнении поиска в ТР-дереве, по сравнению с поиском в Р-дереве.

3.1. Результаты сравнения по размеру дерева и затрат на добавление сведений

Размер созданных индексных структур показан в таблице. Как видно из таблицы, размер ТР-дерева, когда нет объектов для индексации, составляет 6 Мб, в то время как размер трехмерного Р-дерева равен ~19 Кб, т. е. пространственное дерево внутри ТР-дерева заполнено и все отрезки транспортной сети города Oldenbourg уже проиндексированы. Однако, для большого числа объектов размер ТР-дерева намного меньше, чем размер трехмерного Р-дерева. Так, размер ТР-дерева для 2000 движущихся объектов в сети равен приблизительно 11 Мб, в то время как размер соответствующего трехмерного Р-дерева – 38 Мб, поскольку Р-дерево увеличивается каждый раз, когда МО перемещается по дороге (дорожные отрезки сохраняются и индексируются несколько раз каждый раз при движении объекта на том же отрезке). С другой стороны, в ТР-дереве отрезки сохраняются и индексируются только один раз.

Таблица. Размер ТР и Р-деревьев в памяти ПК

	0 объектов		2000 объектов	
	ТР-дерево	Р-дерево	ТР-дерево	Р-дерево
Размер индекса	~6 Мб	~19 Кб	~11 Мб	~38 Мб

* Выбор городов сделан в генераторе сети [9]

3.2. Результаты сравнения по количеству доступов к узлам (вход – выход)

Узел в ТР-дереве соответствует дисковой странице. Доступ к узлу означает запрос чтения или записи на диск. При каждом добавлении объекта в ТР-дерево (для сети г. Oldenbourg) необходимо 8 и 12 узловых записей соответственно, в то время как добавление в трехмерное Р-дерево требует в среднем 4, 5 узловых записей для обеих сетей. Дело в том, что добавление в ТР-дерево требует поиска в двухмерном Р-дереве и затем добавления в одномерное Р-дерево. Таким образом, в ТР-дереве требуется доступ к большему числу узлов, чем в Р-дереве.

3.3. Исследование количества опрашиваемых узлов

Для каждого из ТР-дерева и Р-дерева путем имитации работы сети выяснялось количество узлов в них, участвовавших в реализации этих запросов. При этом были использованы различные размеры окон запросов. Полученные результаты представлены на рис. 4.

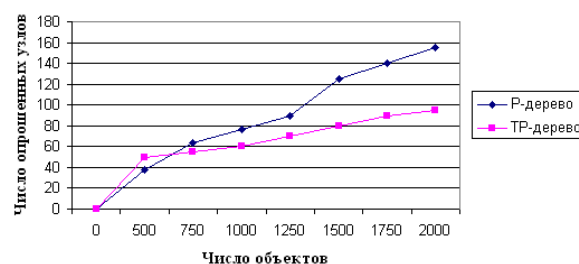


Рис. 4. Зависимость числа опрошенных узлов от числа объектов при традиционных запросах

Как следует из рис. 4, ТР-дерево для небольшого числа (до ~700) движущихся объектов уступает по числу опрошенных узлов трехмерному Р-дереву. Точка пересечения, в которой ТР-дерево имеет такое же число опрошенных узлов, как и Р-дерево, зависит от сети (от числа линейных отрезков) и размера запроса, а также от числа объектов, т. е. в трехмерном Р-дереве размер и стоимость поиска будут расти по мере того, как все больше объектов будет добавляться в дерево.

Заключение

В данной работе мы расширили Р-дерево, чтобы индексировать отрезки траекторий мобильных объектов. Это нужно для обработки запросов о координатах и траекториях мобильных объектов, которые обычно используются группой объектов, как в транспортных сетях. Мы использовали этот факт, чтобы минимизировать размер и мертвое пространство в ТР-дереве. Проведенные эксперименты показали, что при большом числе мобильных объектов предложенное в работе ТР-дерево имеет лучшие показатели по выполнению запроса и обновлению данных, чем трехмерное Р-дерево. Однако, для малого числа объектов предпочтительнее

трехмерное R-дерево. Кроме того, TP-дерево может использоваться для эффективной обработки разных типов запроса о различных промежутках времени от прошлого до будущего. Оно может эффективно использоваться в транспортном контро-

ле и радиовещательных системах. Его внедрение в системы управления базами данных не вызывает сложности, т. к. строится на базе классического пространственного R-дерева, которое поддерживается различными коммерческими СУБД.

СПИСОК ЛИТЕРАТУРЫ

1. Guttman A. R-Trees: a Dynamic Index Structure for Spatial Searching // The ACM-SIGMOD Conference on the Management of Data. — ACM Press, New York, 1984. — P. 47–57.
2. Tao Y., Papadia D.S., Sun J. The TPR*-tree: An optimized spatio-temporal access method for predictive queries // Proc. Intern. Conf. on Very Large Data Bases. — VLDB Endowment, 2003. — P. 790–801.
3. Kollios G., Gunopulos D. Indexing Mobile Objects // The 18th ACM Symposium on Principles of Database Systems. — Philadelphia, 1999. — P. 261–272.
4. Theodoridis Y., Vazirgiannis M., Sellis T. Spatio-temporal indexing for large multimedia applications // The 3rd IEEE Intern. Conf. on Multimedia Computing and Systems. — Hiroshima, Japan, 1996. — P. 441–448.
5. Tzouramanis T., Vassilakopoulos M., Manolopoulos Y. Overlapping Linear Quad Trees: A Spatio-Temporal Access Method // The 6th Intern. Symp. on Advances in Geographic Information Systems. — Bethesda MD, 1998. — P. 1–7.
6. Sellis T., Roussopoulos N., Faloutsos C. The R+ Tree: A Dynamic Index for Multi-Dimensional Objects // In Proc. 13th Intern. Conf. on Very Large Data Bases. — Brighton, England, 1987. — P. 507–518.
7. Hadjieleftheriou M.G., Kollios V.J., Gunopulos D. Efficient indexing of spatiotemporal objects // The 8th Intern. Conf. on Extending Database Technology. — Prague, Czech Republic, 2002. — P. 251–268.
8. Pfoser D., Jensen C.S., Theodoridis Y. Novel approaches to the indexing of moving object trajectories // The 26th Intern. Conf. on Very Large Databases. — Cairo, Egypt, 2000. — P. 395–406.
9. Brinkhoff T. Generating Network-Based Moving Objects // Proc. of the 12th Intern. Conf. on Scientific and Statistical Database Management, SSDBM'00. — Berlin, Germany, 2000. — P. 253–255.

Поступила 18.03.2009 г.

УДК 004.021

РАСШИРЕНИЕ КОНТЕКСТА МОБИЛЬНОГО СЕРВИСА МАРШРУТОМ ПОЛЬЗОВАТЕЛЯ В ПЛАТФОРМЕ STREAMSPIN

Н.А. Шестаков, C.S. Jensen*

Томский политехнический университет

*Университет Ольборга (Aalborg University), Дания

E-mail: nicolai.shestakov@gmail.com

Представлена архитектура платформы мобильных сервисов StreamSpin, использующая открытые интерфейсы для интеграции мобильных сервисов, разработанных внешними провайдерами. Одной из важных особенностей является включение в контекст мобильного сервиса предсказанного маршрута пользователя. Описаны ключевые особенности реализации алгоритма предсказания маршрута пользователя.

Ключевые слова:

LBS-сервис, предсказание маршрута, предиктивная навигация, map matching, context-aware сервис.

Введение

Повсеместное внедрение беспроводных коммуникаций, таких как GPRS или WiFi, а также рост вычислительной мощности мобильных устройств постепенно приводят к росту доли мобильных пользователей в сети Интернет. В связи с этим растет и число мобильных сервисов — услуг, предоставляемых через сеть мобильным пользователям.

Сервисы, использующие информацию о местоположении пользователя, называются сервисами, основанными на местоположении (Location-Based Services, LBS), или LBS-сервисами. Актуальность таких сервисов возрастает с постоянным развитием технологий позиционирования: GSM, GPS/A-GPS,

WiFi и смешанных технологий. Согласно аналитическому докладу компании Berg Insight ожидаемый рост рынка LBS-сервисов составит 50 % в год [1].

Феномен Web 2.0 [2] определил новые концепции в создании и продвижении программных продуктов. Произошло смещение акцентов с десктоп-приложений к сервисам, доступным через сеть. Это видно на примерах развития бизнес-модели «Software-as-a-Service», вычислений в «облаке» («cloud computing») и многих других. При исследовании алгоритмов интересен вопрос, насколько применимы новые подходы и идеи в рамках новой парадигмы, как только эти работы выйдут за пределы тестовых сред и синтетических экспериментов. На-